Einführung in CORE WAR Teil 5

Labels vereinfachen das Programmieren (Forts.)

Wenn man sich die Gewinnchancen von *Hase* betrachtet, muss man feststellen, dass *Hase* zwar manchen Angriff durch die Flucht abwehren kann; aber andererseits bringt er seinen Gegner selten in Gefahr, so dass man ihm gern ein bisschen Aggressivität verleihen möchte. Da *Hase* jetzt unter Verwendung von Labels formuliert wurde, macht es auch keine Probleme mehr, einige Anweisungen einzufügen. Wenn Sie etwa am Ende des Teils Abtasten der Markierung die Anweisungen **`JMP pruef`** durch die folgende Sequenz ersetzen, wird *Hase* nach jedem Abtasten eine Bombe werfen.

```
BombD EQU    503                    ; Abstand der einzelnen Bomben
                                    ; 503 ist Primzahl!

      ADD   #BombD       Bombe      ; Das Bombenziel wird neu eingestellt.
      SLT   #E           Bombe      ; Falls die Bombe hinter Hase liegt,
                                    ; wird die naechste Anweisung uebersprungen.
      JMP    Pruef                  ; Es darf nicht geworfen werden, da sich
                                    ; der Hase sonst selbst trifft.
      SLT   #A-BombD     Bombe      ; Falls das Ziel vor Hase liegt, wird
                                    ; die Bombe geworfen.
      MOV    Bombe      @Bombe
      JMP    Pruef
Bombe DAT   #0          #E+3        ; Bombe mit Ziel in Operand B
```

Die **`SLT`**-Anweisung ist der **`CMP`**-Anweisung verwandt. Sie vergleicht den Wert des *unmittelbaren* Operanden A mit dem Wert, der sich aus dem Operanden B ergibt. Falls der erste kleiner als der zweite Wert ist, wird die nächste Anweisung übersprungen.

Man muß nicht am Anfang beginnen

Es kann vorkommen, dass man mit dem Programmlauf nicht mit der ersten Anweisung beginnen möchte. Dies ist bei folgendem Beispiel der Fall.

```
; Beispiel END-Direktive

Falle MOV    2     -1          ; Knirps-Falle
      JMP    Falle
      DAT   #0    #0
      . . .

Start SLT   <5    @2           ; Hauptteil des Programms
      . . .

      JMP    Falle
      END    Start             ; Das Programm beginnt bei Start
```

Zu Anfang des Programms ist eine *Knirps-Falle* aufgebaut. Wenn *Knirps* in der Speicherzelle **`-2`** bzgl. **`Falle`** steht, kopiert er sich in die **`-1`**. Wenn unser Programm mit ein bisschen Glück unmittelbar anschließend die **`MOV`**-Anweisung bei **`Falle`** ausführt, wird die **`DAT`**-Anweisung von **`+2`** nach **`-1`** kopiert, und das Spiel ist für *Knirps* beim nächsten Zug vorbei.

Wenn man jedoch bei **`Falle`** mit der Programmausführung beginnt, so ist man selbst in die sorgfältig gegrabene Grube gefallen, aus der es kein Entrinnen gibt. Ein einfacher Ausweg ist, hinter der **`END`**-Direktive eine Adresse anzugeben, bei der begonnen werden soll. Diese Adresse wird auch Einsprungadresse genannt. Es können dort Ausdrücke wie bei der **`EQU`**-Direktive verwendet werden.

Quelle: Aus dem Begleitheft zu Core War – Krieg der Kerne, Spektrum der Wissenschaft 1993, S. 32 f.